ABSTRACT OF THE DISCLOSURE

The invention provides a method and system for reliably performing extra-long operations in a reliable state-full system (such as a file system). The system records consistency points, or otherwise assures reliability, notwithstanding the continuous performance of extra-long operations and the existence of intermediate states for those extra-long operations. Moreover, performance of extra-long operations is both deterministic and atomic with regard to consistency points (or other reliability techniques used by the system). The file system includes a separate portion of the file system reserved for files having extra-long operations in progress, including file deletion and file truncation. This separate portion of the file system is called the zombie filespace; it includes a separate name space from the regular ("live") file system that is accessible to users, and is maintained as part of the file system when recording a consistency point. The file system includes a file deletion manager that determines, before beginning any file deletion operation, whether it is necessary to first move the file being deleted to the zombie filespace. The file system includes a zombie file deletion manager that performs portions of the file deletion operation on zombie files in atomic units. The file system also includes a file truncation manager that determines, before beginning any file truncation operation, whether it is necessary to create a complementary file called an "evil twin". The truncation manager will move all blocks to be truncated from the file being truncated to the evil twin file. The file system includes a zombie file truncation manager that performs portions of the file truncation operation on the evil-twin file in atomic units. An additional

advantage provided by the file system is that files having attached data elements, called "composite" files, can be subject to file deletion and other extra-long operations in a natural and reliable manner. The file system moves the entire composite file to the zombie filespace, deletes each attached data element individually, and thus resolves the composite file into a non-composite file. If the non-composite file is sufficiently small, the file deletion manager can delete the non-composite file without further need for the zombie filespace. However, if the non-composite file is sufficiently large, the file deletion manager can delete the non-composite file using the zombie filespace.